PSIM Tutorial

Auto Code Generation for F2803x Target

February 2013

With SimCoder and the F2803x Hardware Target, PSIM can automatically generate ready-to-run code from the control schematic for hardware based on TI F2803x series fixed-point DSP.

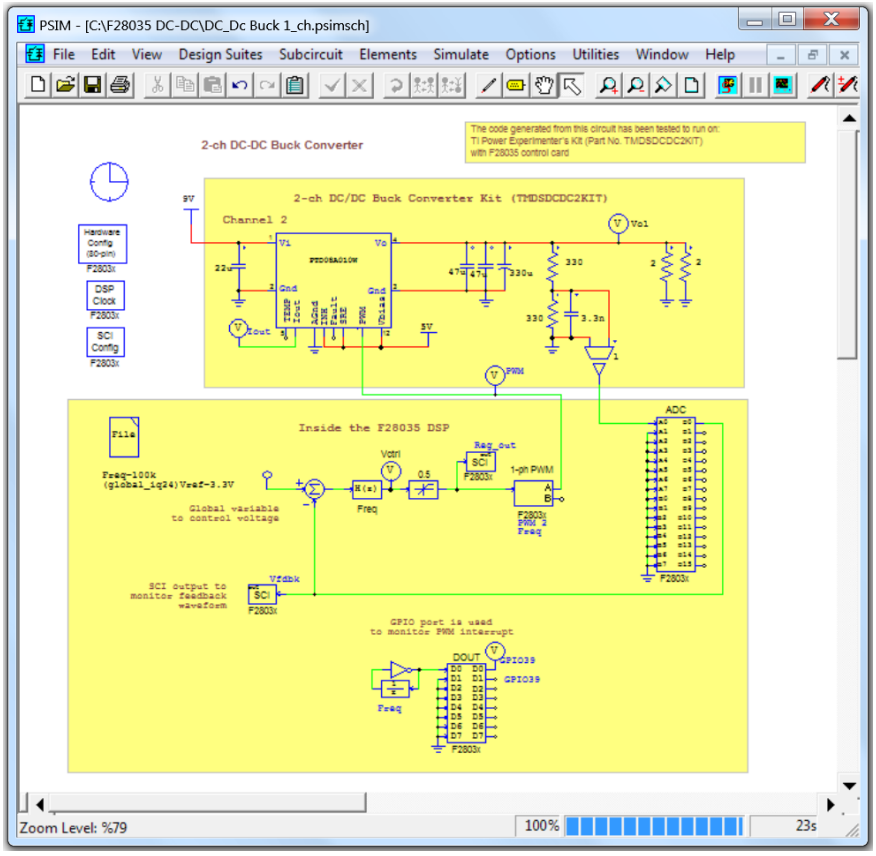
This tutorial describes, in step by step, how to generate code in PSIM, compile and upload the code in CodeComposerStudio, and run it on the DSP.

To illustrate the process, we use the circuit "*DC_DC Buck 1_ch.psimsch*" as an example. This example is located in the sub-folder "Examples\SimCoder\F2803x Target\TI 1-Ch DC-DC with SCI" in the PSIM directory.

To keep the original example unchanged, we will copy the whole folder to "C:\ F28035 DC-DC", and use this folder as the working folder in this tutorial.

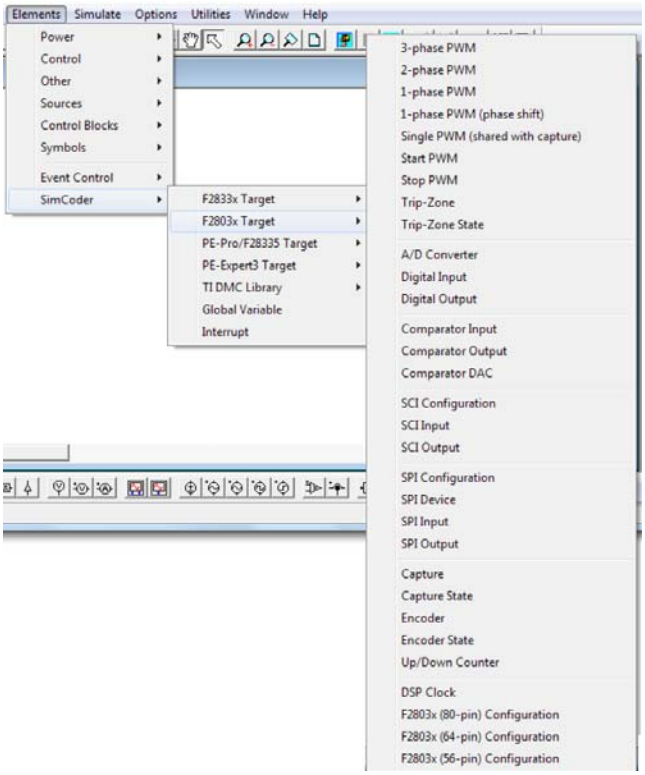
1. Setting Up in PSIM

In PSIM, load the example schematic file "*DC_DC Buck 1_ch.psimsch*" as shown below.


PSIM - [C:\F28035 DC-DC\DC_Dc Buck 1_ch.psimsch]
File Edit View Design Suites Subcircuit Elements Simulate Options Utilities Window Help
2-ch DC-DC Buck Converter
2-ch DC/DC Buck Converter Kit (TMDSDCDC2KIT)
Channel 2
Inside the F28035 DSP
Zoom Level: %79
100%

This example generates the code for the TI 2-Channel DC/DC Buck Converter Kit (TMDSDCDC2KIT) with a Piccolo F28035 controlCARD. Only one channel is implemented in this example. One can simulate this circuit by selecting **Simulate** >> **Run Simulation**.

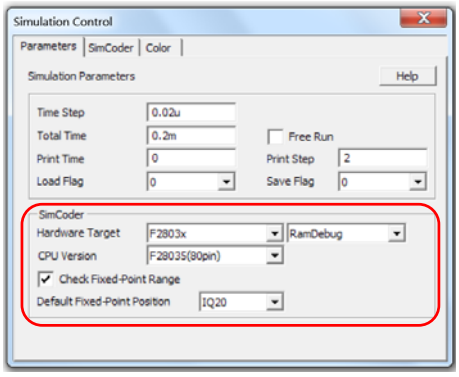
For fixed-point code generation, a circuit must contain elements from the F2803x Hardware Target library. The library can be accessed by going to **Elements** >> **SimCoder** >> **F2803x Target**. A list of the library is shown below.



1.1 Setting Up Simulation Control

Before the code is generated, SimCoder parameters must be properly set in Simulation Control.

Double click on the Simulation Control block (the clock image), and set the parameters of SimCoder the dialog block opens as shown below.



The settings are:

- *Hardware Target*: set to *F2803x* and *RamDebug*. RamDebug is selected in this example for easy debugging.
- CPU Version: select F2803 (80pin).
- *Check Fixed-Point Range*: check this box to check the fixed-point range for overflow.
- Default Fixed-Point Position: select IQ20 for this example.

Currently, project files can be generated for CCS v3.3 only. If one uses a version newer than v3.3, one needs to convert the v3.3 project.

1.2 Defining F2803x Target Elements

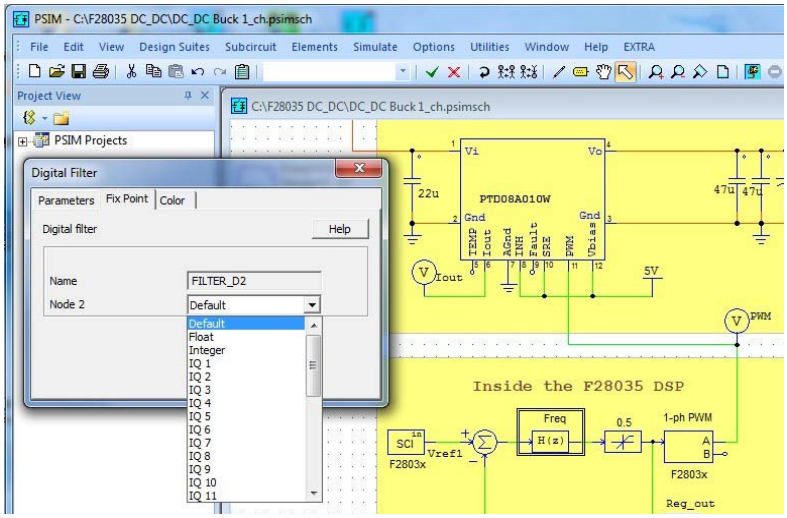
In this circuit, there are several F2803x Target library elements. All these elements must be properly configured according to the specific target hardware settings.

- *F2803x Hardware Configuration*: This element defines the I/O ports of the F2803x hardware. One must Unlock this element to change the designation of the I/O port, and must Lock it afterwards. In this example:
  - A0 is designated as an A/D input channel for voltage measurement;
  - GPIO2 is designated as PWM output;
  - GPIO28 and GPIO29 are designated for SCI communication; and
  - GPIO39 is used as a digital output.
- *DSP Clock*: Specify the DSP clock parameters for this project as follows:
  - *DSP Clock Source*: Internal oscillator 1
  - *External Clock* (*MHz*): 10
  - *DSP Speed* (*MHz*): 60
- *SCI Configuration*: Specify the I/O pins, the speed, and parity check of the SCI port. The output buffer size is determined by the desired waveform resolution and available memory space. The SCI communication provides a convenient way to utilize PSIM's DSP Oscilloscope function. The DSP Oscilloscope is used to control the buck converter's output voltage and monitor the values inside the DSP. The SCI input allows the reference value to be changed by the DSP Oscilloscope in real time, and SCI outputs are used to display signals it the DSP Oscilloscope in real time.
- *A/D Converter* (16 channel): Specify the mode and gain of ADC channels. For this example, Channel A0 is in DC mode with a gain of 2.0
- *PWM Module* (single phase): Specify the PWM module source, the output mode, switching frequency, and many other settings of a single-phase PWM module. For further detailed information, please refer to the SimCoder User Manual.

1.3 Defining Fixed-Point Positions

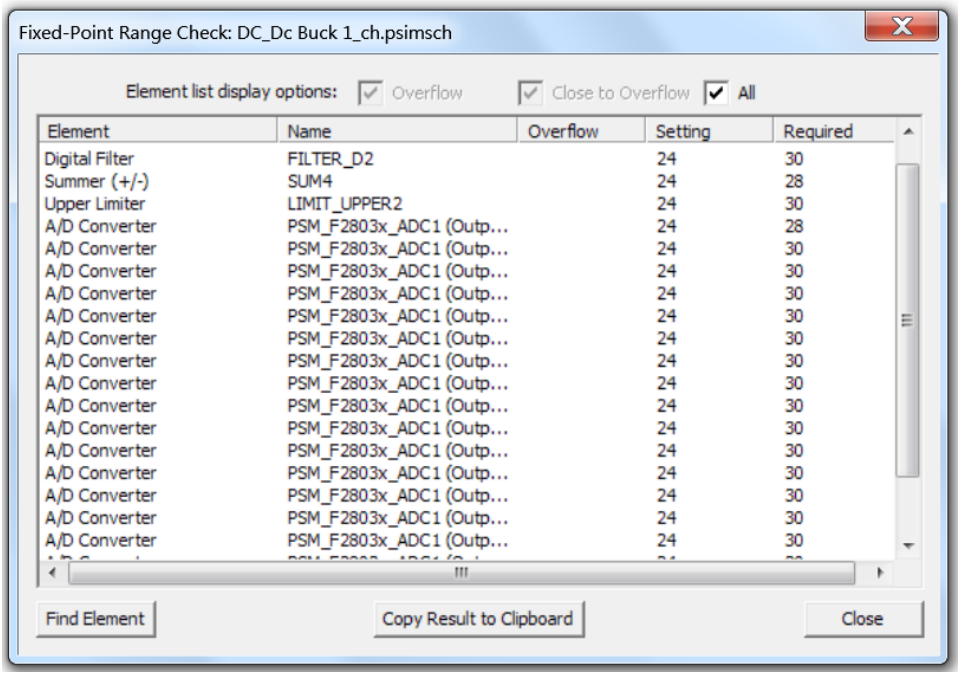
In the F2803x Target code, Texas Instruments' IQmath library is used for fixed-point calculation. As shown in the figure below, each of the non-hardware-related elements can be set as *Default*, *Integer*, or fixed point with the IQ number from 1 to 30.

In this example, the *Default* fixed-point position is selected for the digital filter block FILTER_D2, which means that the output of this block will use the fixed-point position of IQ20 as defined in *Default Fixed-Point Position* in Simulation Control.



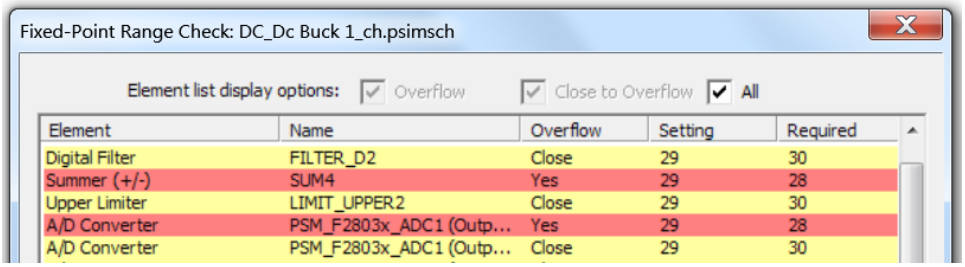
In fixed-point calculation, the selection of the fixed-point position is an important issue. If the fixed-point range is too low, the variable value may overflow. But if the fixed-point range is too high, the calculation resolution will suffer. To address this problem, PSIM provides the function to check the fixed-point range from simulation results, and give indication if a particular output is close to overflow or already overflow, or is too conservative in the fixed-point position setting.

In this example, after the simulation, the following dialog window will appear.



The dialog window shows that, for example, the output of the digital filter FILTER_D2 is set to IQ24, and the required setting is IQ30, which is more than sufficient in the data range.

However, as an example, if the default fixed-position is set to IQ29, it will not be sufficient as some of the variables will overflow and others are close to overflow. The fixed-point range check results are shown below:

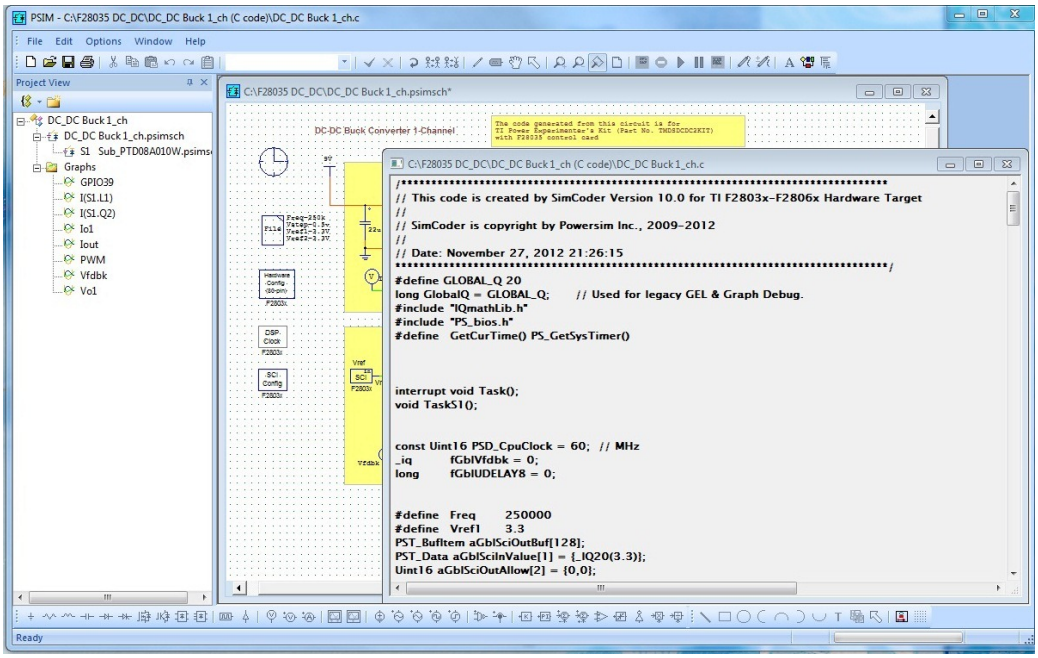


The results show that the summer and A/D converter outputs will overflow, and the digital filter and limiter outputs are close to overflow.

PSIM's range check function provides a convenient way to handle fixed-point positions.

1.4 Generating Hardware Code

To generate code, select **Simulate** >> **Generate Code**. The generated code will be displayed in a separate window, as shown below.



PSIM will create a subfolder in the same folder as the PSIM circuit file and is named the same as the PSIM circuit but with (C code) added at the end. For this project, this subfolder is C:\F28035 DC_DC\DC_DC Buck 1_ch (C code).

In addition to the C code file, PSIM also generates project files, link command files, as well as all other files necessary for four different project building configurations: RAM Debug, RAM Release, Flash Release, and Flash RAM Release. All these files are stored in the newly created subfolder C:\F28035 DC_DC\DC_DC Buck 1_ch (C code).

2. Setting Up Hardware

Insert the TMS320F28035 controlCARD into the 2-Channel DC/DC Buck Converter Kit (TMDSDCDC2KIT), as shown below.

Connect the JTAG emulator between the J1 connector of the board and one of the computer's USB ports. After connecting the power to the board, turn on Switches SW1 and SW2 on the board. Switch SW3 is for displaying either Ch1 or Ch2 output. Set SW3 to the Ch2 position.

3. Loading the Code to Hardware Using CCS

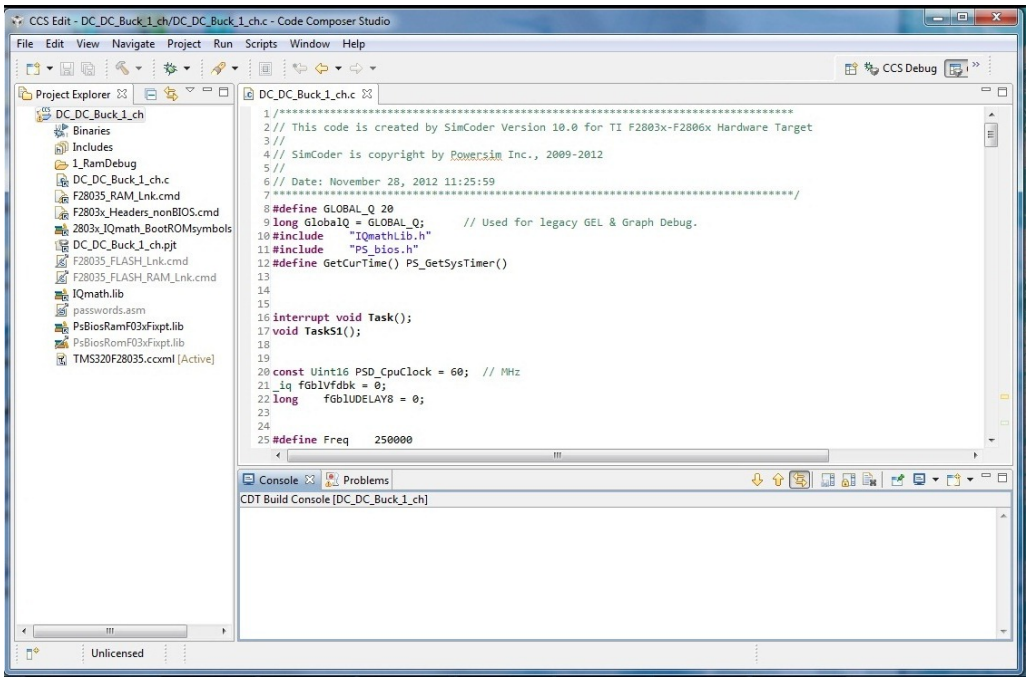
Currently, project files are generated for CCS v3.3 only. For a version higher than v3.3, the project file must be converted.

3.1 Connecting Hardware with CCS v5

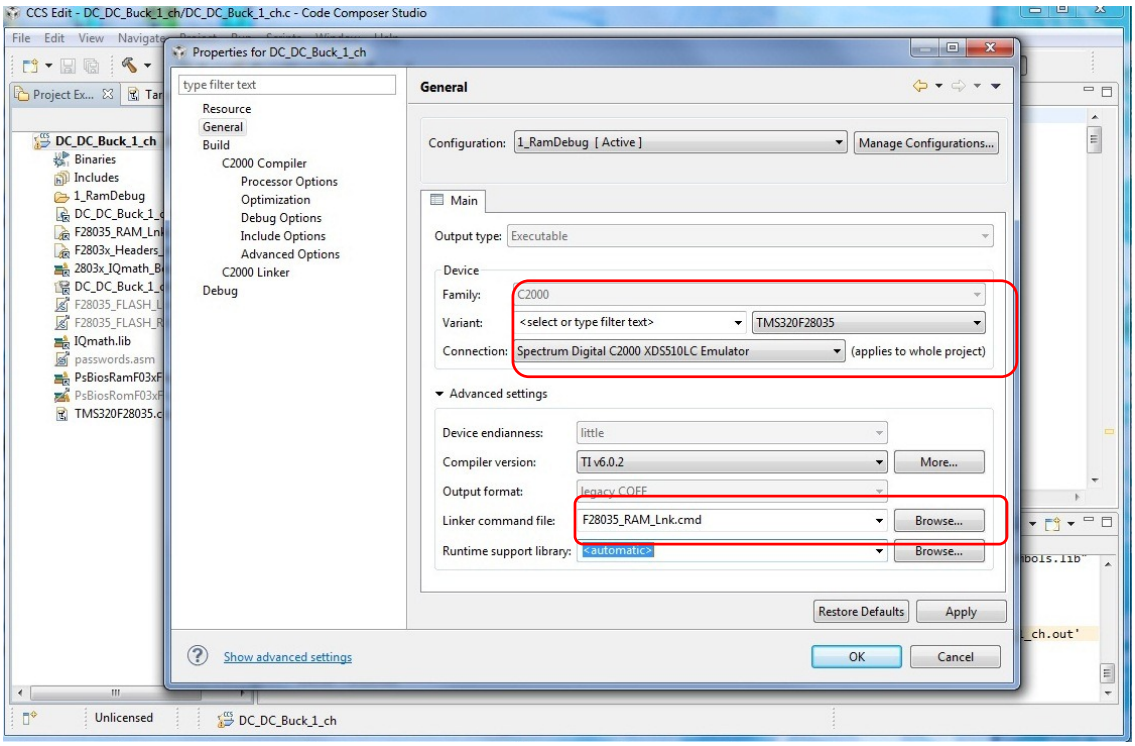
The Code Composer Studio used in this section is Version 5.1.1.00031.

Launch CCS v5. When prompted to select a workspace, click the **Browse** button to create a new folder for the files generated by CCS. For this example, a new folder named CCS workspace is created inside the F28035 DC_DC folder.

- Select **Project** >> **Import Legacy CCSv3.3 Project**. Navigate to the folder "C:\ F28035 DC-DC\DC_DC Buck 1_ch (C code)". Select the project file "DC_DC Buck 1_ch.pjt". The CCS Editor will be shown as below.



- Select **Project** >> **Properties**. Set the properties of the project DC_DC_Buck 1_ch as shown below.

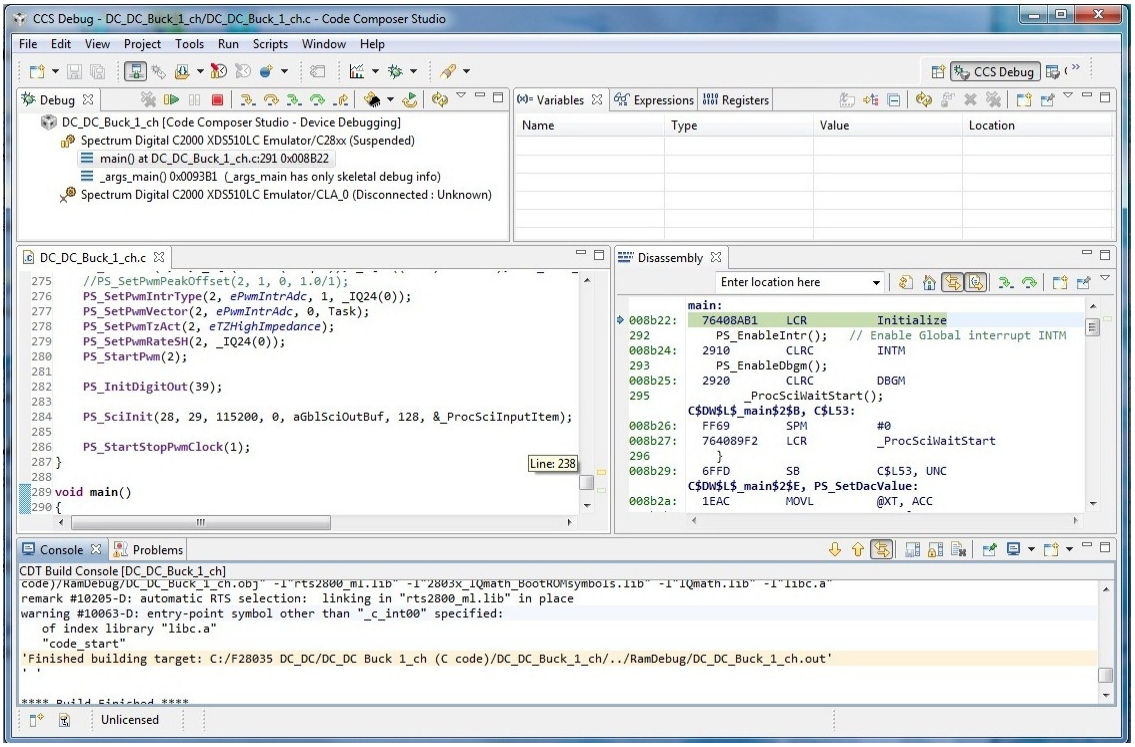


The settings are:

- *Configuration*: RamDebug (Active)
- *Device Family*: C2000
- *Device Variant*: TMS320F28035
- *Connection*: Select the correct connection from the list, according to the JTAG connection you have.

In the *Advanced settings*, use the browser to find the file F28035_RAM_Lnk.cmd in the folder C:\F28035 DC_DC\DC_DC Buck 1_ch (C code).
Click **Apply**, and then **OK**.

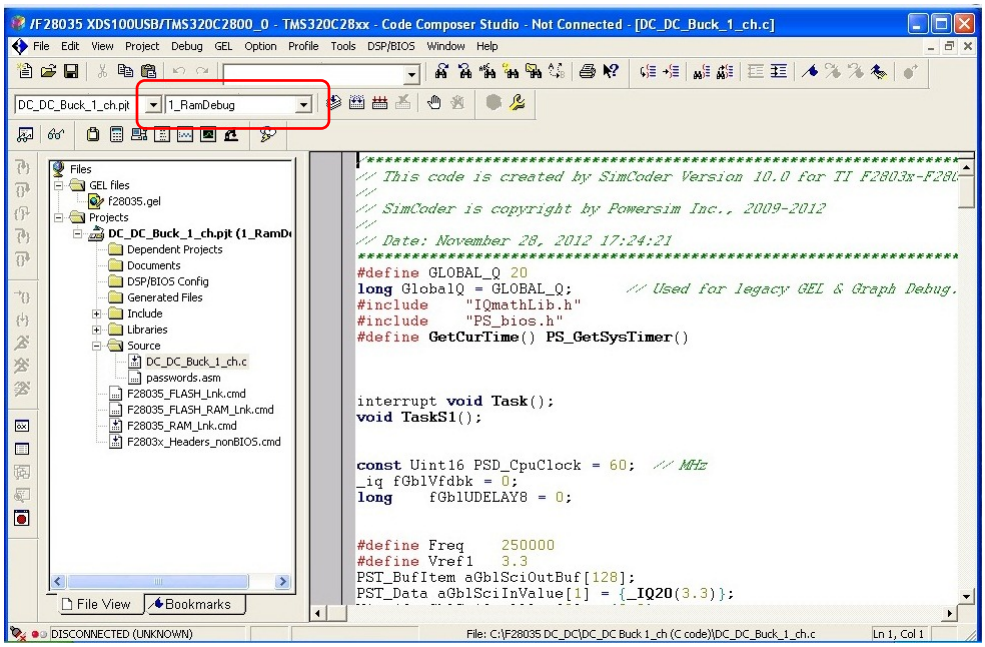
- Select **Run** >> **Debug**. The hardware target will be connected, and the CCS will display the Debugger window as shown in the figure below. The code is ready to run in targeted DSP hardware.
- Select **Run** >> **Resume** to resume the code execution. The LED display of the EVM would display the DC-DC converter output voltage at 3.3V (+/- 0.05V).
- Section 4 of this tutorial will illustrate how to use DSP Oscilloscope to monitor the DSP internal variables and to control the converter voltage.



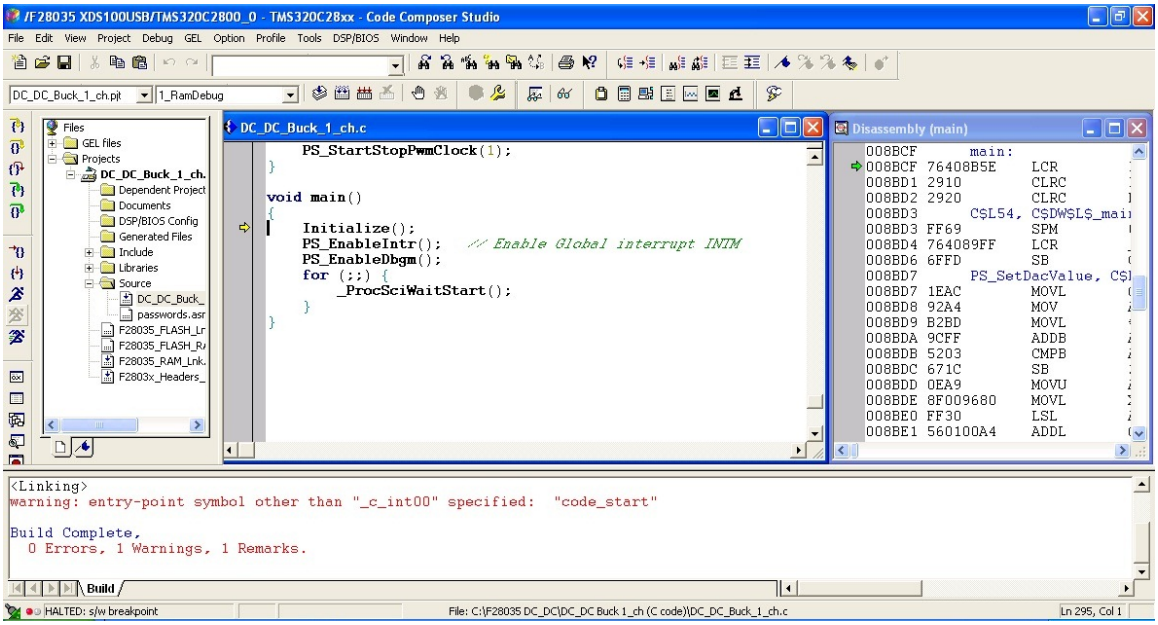
3.2 Connecting Hardware with CCS V3.3

Launch CCS v3.3 Setup to create a system configuration for the TMS320F28035 controlCARD and proper JTAG emulator. Launch CCS v3.3, select the correct system configuration.

- Go to **Project >> Open**. Navigate to the folder "c:\ F28035 DC-DC\DC_DC Buck 1_ch (C code)". Select the project file "DC_DC Buck 1_ch.pjt. The CCS window will appear as shown below.



- Set *Active Configuration* to *RAM Debug*. All the program and data will be loaded to the RAM memory.
- To compile the project, select **Project >> Build** to build the project (or select **Project >> Rebuild All** to rebuild the project).
- Select **Debug >> Connect** to connect the computer to the DSP.
- Go to **File >> Load Program** and locate the .out file (in this example "c:\ F28035 DC-DC\DC_DC Buck 1_ch (C code)\RamDebug\DC_DC_Buck 1_ch.out"). CCS will automatically load the code into the RAM memory. Note that the Load Program function works only for the RAM Debug or RAM Release version.
- Select **Debug >> Reset CPU**. Then select **Debug >> Restart**, and **Debug>> Go Main**. The cursor will indicate the beginning of the main program, as shown below.



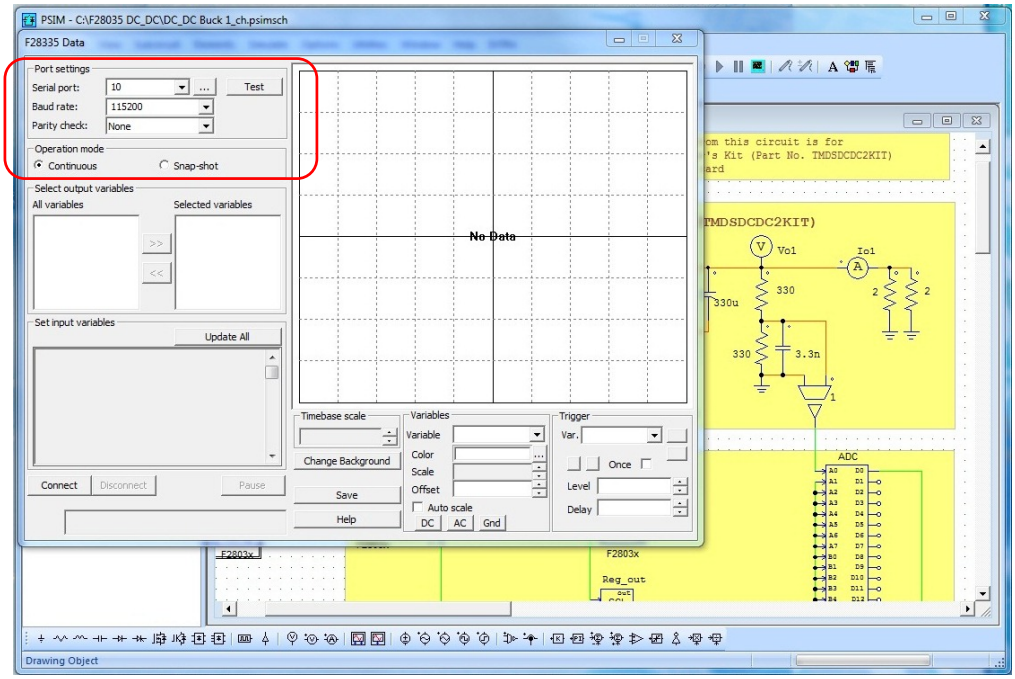
- Select **Debug >> Run** to run the code.

Section 4 of this tutorial will demonstrate how to use DSP Oscilloscope to monitor DSP internal variables and to control the converter voltage.

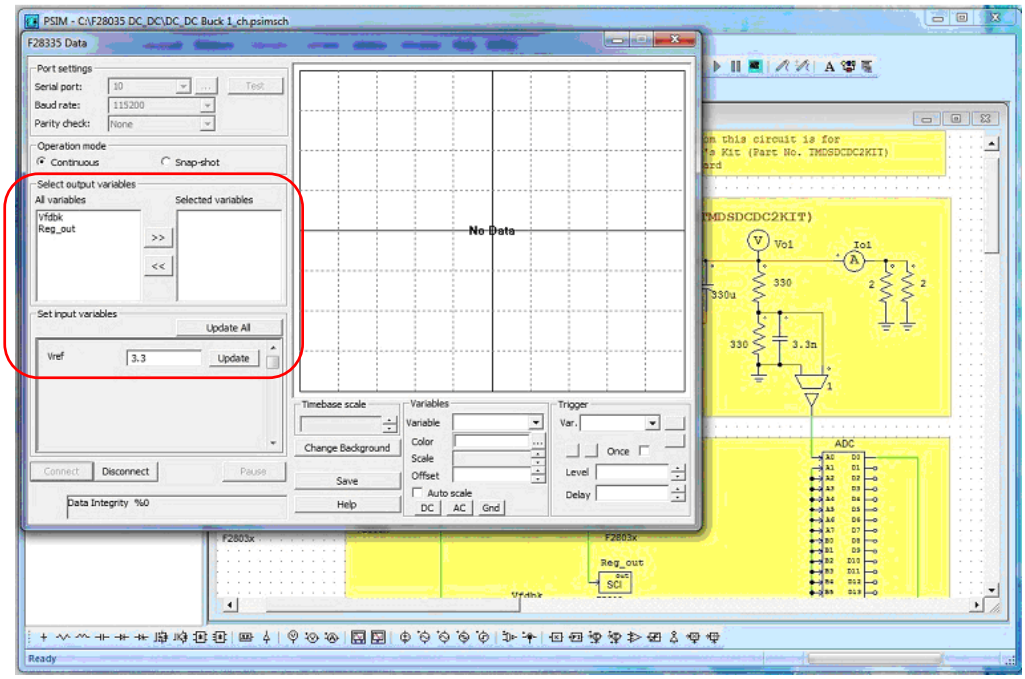
4. Monitoring DSP Waveforms with DSP Oscilloscope

At the end of Sections 3.1 and 3.2, the code is running in the targeted DSP to generate a converter output of 3.3V (+/-0.05V). Now, one may use PSIM's DSP Oscilloscope feature to monitor the variables inside the DSP and to control the converter output voltage.

- Connect the SCI port of the hardware target to the computer.
- Launch PSIM. Select **Utilities** >> **DSP Oscilloscope**. This will launch the DSP Oscilloscope as shown below.

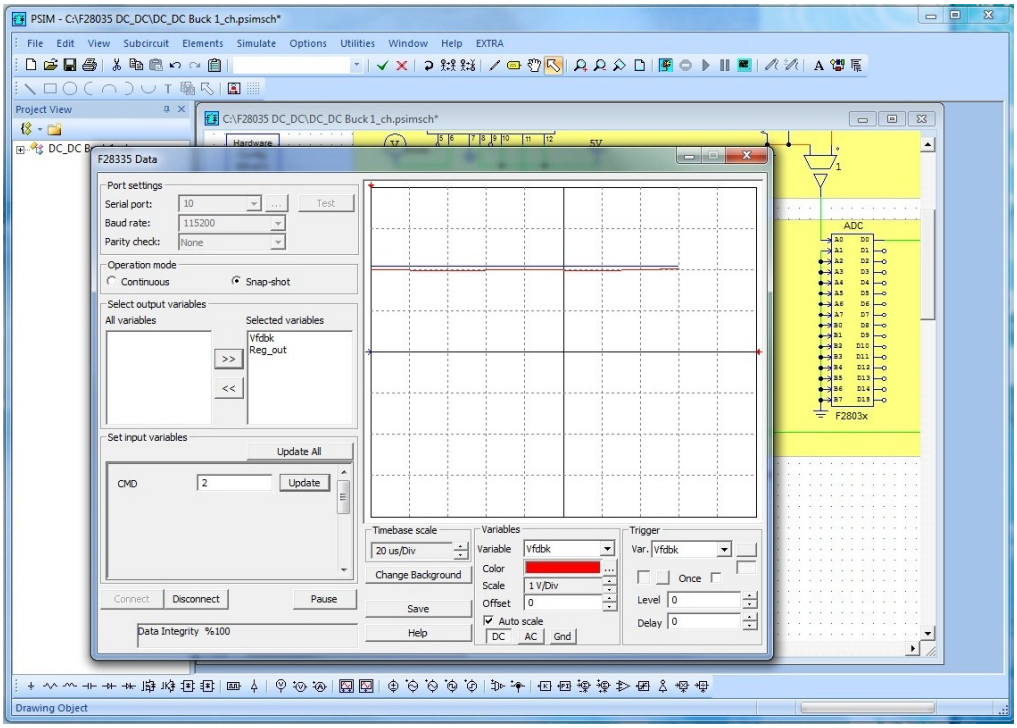


- Set the correct serial port number, baud rate, and parity check. They must be identical to these in the SCI Configuration block in the PSIM circuit.
- Click the Connect button at the left bottom of the scope panel. All names of SCI output and input variables will be listed on the left side of the panel, as shown below.



The two variables available for monitoring are Vfdbk and Ref_out. Select the variables to display on the scope screen.

- To change the DC-DC converter output voltage, modify the value Vref to 2 and click the **Update** button. The LED display of the converter board will change to the new value. The figure below shows the waveforms with the new value of Vref.



5. Code Composer Studio References

For further information about the Texas Instrument's Code Composer Studio, please refer to the documents in the links below.

- CCSv5 Getting Started GuideCCSv5 Getting Started Guide

  http://processors.wiki.ti.com/index.php/CCSv5_Getting_Started_Guide

- Code Composer Studio Development Tools v3.3 - Getting Started Guide

  http://www.ti.com/lit/ug/spru509h/spru509h.pdf